Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method in a data processing system for managing a set of processors, the method comprising:

receiving a call from an operating system, wherein the call indicates that a selected processor in the set of processors is unneeded for a period of time that is included in the call from the operating system, wherein the call is a sub-processor partitioning call sent from one of a plurality of different operating systems which are each executing in a respective partition of a multi-partitioned data processing system, where each partition of the multi-partitioned data processing system is managed by a runtime abstraction layer and (i) is assigned a non-overlapping subset of resources of the data processing system, and (ii) can be individually booted and shut down without having to power-cycle entirety of the data processing system; and

altering operation of the selected processor to reduce power usage during the period of time, wherein the receiving step and the altering step are performed by the runtime abstraction layer that receives the call from at least one of a plurality of operating systems that are concurrently executing in the data processing system.

2. (Cancelled)

3. (Previously Presented) The method of claim 1, wherein the period of time is specified in the call from the operating system and is a time during which idle cycles are present for the selected processor.

4. (Original) The method of claim 1, wherein the selected processor is in an original state prior altering operation of the selected processor and further comprising:

returning the selected processor to the original state after the period of time has elapsed.

5. (Original) The method of claim 4 further comprising:

returning the selected processor to the original state if the period of time has not elapsed and an external interrupt indicating work is present for the selected processor is received.

6. (Cancelled)

7. (Original) The method of claim 1, wherein the altering step comprises:

reducing a clock speed of the selected processor.

8. (Previously Presented) The method of claim 1, wherein the altering step comprises:

placing the selected processor in a sleep mode if the period of time is greater than a predetermined threshold.

9. (Currently Amended) A data processing system for managing a set of processors, the data processing system comprising:

receiving means for receiving a call from an operating system, wherein the call indicates that a selected processor in the set of processors is unneeded for a period of time that is included in the call from the operating system, wherein the call is a sub-processor partitioning call sent from one of a plurality of different operating systems which are each executing in a respective partition of a multi-partitioned data processing system, where each partition of the multi-partitioned data processing system is managed by a runtime abstraction layer and (i) is assigned a non-overlapping subset of resources of the data processing system, and (ii) can be individually booted and shut down without having to power-cycle entirety of the data processing system; and

altering means for altering operation of the selected processor to reduce power usage during the period of time, wherein the receiving means and the altering means are located in the runtime abstraction layer that receives the call from at least one of a plurality of operating systems that are concurrently executing in the data processing system.

10. (Cancelled)

11. (Previously Presented) The data processing system of claim 9, wherein the period of time is specified in the call from the operating system and is a time during which idle cycles are present for the selected processor.

12. (Original) The data processing system of claim 9, wherein the selected processor is in an original state prior altering operation of the selected processor and further comprising:

returning means for returning the selected processor to the original state after the period of time has elapsed.

13. (Original) The data processing system of claim 12, wherein the returning means is a first returning means and further comprising:

second returning means for returning the selected processor to the original state if the period of time has not elapsed and an external interrupt indicating work is present for the selected processor is received.

14. (Cancelled)

15. (Original) The data processing system of claim 9, wherein the altering means comprises:

reducing means for reducing a clock speed of the selected processor.

16. (Previously Presented) The data processing system of claim 9, wherein the altering means comprises:

placing means for placing the selected processor in a sleep mode if the period of time is greater than a predetermined threshold.

17. (Currently Amended) A computer program product encoded in a computer ~~readable medium~~ recordable-type media and executable by a data processing system for managing a set of processors, the computer program product comprising:

first instructions for receiving a call from an operating system, wherein the call indicates that a selected processor in the set of processors is unneeded for period of time that is included in the call from the operating system, wherein the call is a sub-processor partitioning call sent from one of a plurality of different operating systems which are each executing in a respective partition of a multi-partitioned data processing system, where each partition of the multi-partitioned data processing system is managed by a runtime abstraction layer and (i) is assigned a non-overlapping subset of resources of the data processing system, and (ii) can be individually booted and shut down without having to power-cycle entirety of the data processing system; and

second instructions for altering operation of the selected processor to reduce power usage during the period of time, wherein the first instructions and the second instructions are located in the runtime abstraction layer that receives the call from at least one of a plurality of operating systems that are concurrently executing in the data processing system.

18. (Cancelled)

19. (Previously Presented) The computer program product of claim 17, wherein the period of time is specified in the call from the operating system and is a time during which idle cycles are present for the selected processor.

20. (Original) The computer program product of claim 17, wherein the selected processor is in an original state prior altering operation of the selected processor and further comprising:

third instructions for returning the selected processor to the original state after the period of time has elapsed.

21. (Original) The computer program product of claim 20 further comprising:

fourth instructions for returning the selected processor to the original state if the period of time has not elapsed and an external interrupt indicating work is present for the selected processor is received.

22. (Cancelled)

23. (Original) The computer program product of claim 17, wherein the second instructions comprises:

sub-instructions for reducing a clock speed of the selected processor.

24. (Previously Presented) The computer program product of claim 17, wherein the second instructions comprises:

sub-instructions for placing the selected processor in a sleep mode if the period of time is greater than a predetermined threshold.

25. (Currently Amended) A data processing system comprising:

a bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and a processing unit connected to the bus system, wherein the processing unit executes a set of instructions to receive a call from an operating system, wherein the call indicates that a selected processor in [[the]] a set of processors is unneeded for a period of time as specified in the call from the operating system; and alter operation of the selected processor to reduce power usage during the period of time, wherein the call is a sub-processor partitioning call sent from one of a plurality of different operating systems which are each executing in a respective partition of a multi-partitioned data processing system, where each partition of the multi-partitioned data processing system is managed by a runtime abstraction layer and (i) is assigned

a non-overlapping subset of resources of the data processing system, and (ii) can be individually booted and shut down without having to power-cycle entirety of the data processing system, and wherein the set of instructions are located in the runtime abstraction layer that receives the call from at least one of a plurality of operating systems that are concurrently executing in the data processing system.